

spacewar 4.0ts 5/4/63 ddp •pt 1

```

3/      jmp sbf          / ignore seq. break
        jmp a40
        jmp a1          / use test word for control, not iot 11 co

```

```

/ interesting and often changed constants

```

```

/symb loc usual value (all instructions are executed,
/ and may be replaced by jda or jsp)

```

```

tno, 6, law i 41          / number of torps + 1
tv1, 7, sar 4s          / torpedo velocity
rlt, 10, law i 20        / torpedo reload time
tlf, 11, law i 140       / torpedo life
foo, 12, -20000          / fuel supply
maa, 13, 10             / spaceship angular acceleration
sac, 14, sar 4s          / spaceship acceleration
str, 15, 100            / star capture radius
me1, 16, 6000           / collision "radius"
me2, 17, 3000           / above/2
ddd, 20, -0             / 0 to save space for ddt
the, 21, sar 9s          / amount of torpedo space warpage
mhs, 22, law i 10        / number of hyperspace shots
hd1, 23, law i 40        / time in hyperspace before breakout
hd2, 24, law i 100       / time in hyperspace breakout
hd3, 25, law i 200       / time to recharge hyperfield generators
hr1, 26, scl 9s          / scale on hyperspatial displacement
hr2, 27, scl 4s          / scale on hyperspatially induced velocity
hur, 30, 40000          / hyperspatial uncertainty
ran, 31, 0              / random number
grv, 32, sar 6s          / gravitational constant

```

```

/ place to build a private control word routine.
/ it should leave the control word in the io as follows.
/ high order 4 bits, rotate ccw, rotate cw, (both mean hyperspace)
/   fire rocket, and fire torpedo. Low order 4 bits, same for
/   other ship. Routine is entered by jsp cwg.

```

```

40/

```

```

cwr,      jmp mg1        / normally iot 11 control
. 20/     / space

```

```

ioh=xct (nop          /delay for dpy's

```

/ routine to flush sequence breakes, if they occur.

```

sbf,      tyi
          lio 2
          lac 0
          lsm
          jmp i 1

```

```

          define
xincr X,Y,INS
          lac Y
          INS ssn
          dac Y
          lac X
          INS scn
          dac X
          term

```

```

          define
yincr X,Y,INS
          lac Y
          INS scn
          dac Y
          lac X
          -INS+add+sub ssn
          dac X
          terminate

```

```

          define
dispatch
          add (. 3
          dap . 1
          jmp .
          term

```

```

          define
dispt A,Y,B
          repeat 6 B=B+B
          lio Y
          dpy-A+B
          term

```

```

          define
scale A,B,C
          lac A
          sar B
          dac C
          term

```

```
define
diff V,S,SF
  add i V
  dac i V
  xct SF
  add i S
  dac i S
  term
```

```
define
random
  lac ran
  rar 1s
  xor (355670
  add (355670
  dac ran
  term
```

```
define
ranct S,SS,C
  random
  S
  SS
  sma
  cma
  dac C
  terminate
```

```
define
    varsft
    dzm xys
    dac t1
    idx xys
v2,    idx xys
    lac t1
    scr 2s
    dac t1
    sza
    jmp v2+R
    scr 2s
    swap
    terminate
```

```
define
    undosft
    dac t1
    dio t2
    lac xys
    add sft
    dap .+1
    lac .
    dac .+6
    dac .+6
    xor (10000
    dac xyt
    lac t1
    dio t2
    scr .
    scr .
    terminate
```

/ change scr to scl or scl to scr.

```
define
    integrate A,B
    cli
    lac i A
    scr 9s
    scr 1s
    div t1
    hlt
    cma+cli-opr
    xct xyt
    xct grv
    dac B
    terminate
```

```
sft,    lac .-1
    scr 7s
    scr 6s
    scr 5s
    scr 4s
    scr 3s
    scr 2s
    scr 1s
    scr
    scl 1s
```

/sine-cosine subroutine•Adams associates
 /calling sequence= number in AC, jda sin or jdacos.
 /argument is between +2 pi, with binary point to right of bit 3.
 /answer has binary point to right of bit 0. Time = 2.35-? ms.
 /changed for auto-multiply , ddp 1/19/63

```

cos,      0
          dap csx
          lac (62210
          add cos
          dac sin
          jmp .+4

sin,      0
          dap csx
          lac sin
          spa
si1,      add (311040
          sub (62210
          sma
          jmp si2
          add (62210

si3,      ral 2s
          mul (242763
          dac sin
          mul sin
          dac cos
          mul (756103
          add (121312
          mul cos
          add (532511
          mul cos
          add (144417
          mul sin
          scl 3s
          dac cos
          xor sin
          sma
          jmp csx-1
          lac (377777
          lio sin
          spi
          cma
          jmp csx

          lac cos
csx,      jmp .

si2,      cma
          add (62210
          sma
          jmp si3
          add (62210
          spa
          jmp .+3
          sub (62210
          jmp si3

          sub (62210
          jmp si1
  
```

```

/integer square root
/input in ac, binary point to right of bit 17, jda sqt
/answer in ac with binary point between bits 8 and 9
/largest input number = 177777

```

```

sqt,      0
          dap sqx
          law i 23
          dac sq1
          dzm sq2
          lio sqt
          dzm sqt

sq3,      isp sq1
          jmp .+3
          lac sq2
sqx,      jmp .

          lac sq2
          sal 1s
          dac sq2
          lac sqt
          rcl 2s
          sza i
          jmp sq3
          dac sqt
          lac sq2
          sal 1s
          add (1
          sub sqt
          sma+sza-skp
          jmp sq3
          spa
          cma
          dac sqt
          idx sq2
          jmp sq3

sq1,      0
sq2,      0

```

```

/outline compiler
/ac=where to compile to, call      jda oc
                                   /ot=address of outline table

```

```
define
```

```

    plinst A
    lac A
    dac i oc
    idx oc
    terminate

```

```
define
```

```

    comtab A, B
    plinst A
    jsp ocs
    lac B
    jmp oce
    terminate

```

```

ocs,      dap ocz                /puts in swap
          dio i oc
          idx oc
          dio i oc
          idx oc
ocz,      jmp .

```

```

oc,        0                    /outline compiler proper
          dap ocx
          lac i ocx
          dap ocg
          plinst (stf 5
          dap ocm
          idx ocx

```

```

ock,      plinst (lac sx1
          plinst (lio sy1
          clf 6

```

```

ocj,      setup occ,6
ocg,      lio .                /outline table
och,      cla

```

```

          rcl 3s
          dio oc1
          lio (rcl 9s
          dispatch
          opr

```

```

          jmp oc1
oco,      jmp oc2
ocq,      jmp oc3
ocp,      jmp oc4
ocr,      jmp oc5
          jmp oc6

```

```

plinst (szf 5          /7 code
add (4
dap ocn
plinst ocn
plinst (dac  $\overline{s}x1$ 
plinst (dio  $\overline{s}y1$ 
plinst (jmp sq6
plinst (clf  $\overline{5}$ 
plinst (lac  $\overline{s}cm$ 
plinst (cma  $\overline{\quad}$ 
plinst (dac  $\overline{s}cm$ 
plinst (lac  $\overline{s}sm$ 
plinst (cma  $\overline{\quad}$ 
plinst (dac  $\overline{s}sm$ 
plinst (lac  $\overline{c}sm$ 
plinst (lio  $\overline{s}sd$ 
plinst (dac  $\overline{s}sd$ 
plinst (dio  $\overline{c}sm$ 

plinst (lac  $\overline{s}sc$ 
plinst (lio  $\overline{c}sn$ 
plinst (dac  $\overline{c}sn$ 
plinst (dio  $\overline{s}sc$ 
plinst ocm
ocx,    jmp .

ocm,    jmp .
ocn,    jmp .

oc1,    plinst (add  $\overline{s}sn$ 
        jsp ocs
        lac (sub  $\overline{s}cn$ 
oce,    dac i oc
        idx oc
        jsp ocs
        plinst (ioh
        lac (dpy-4000
ocd,    dac i oc
        idx oc
        lio  $\overline{oc}i$ 
        count  $\overline{oc}c$ , och
        idx  $\overline{oc}g$ 
        jmp  $\overline{oc}j$ 

oc2,    comtab (add  $\overline{s}cm$ , (add  $\overline{s}sm$ 
oc3,    comtab (add  $\overline{s}sc$ , (sub  $\overline{c}sm$ 
oc4,    comtab (sub  $\overline{s}cm$ , (sub  $\overline{s}sm$ 
oc5,    comtab (add  $\overline{c}sn$ , (sub  $\overline{s}sd$ 
oc6,    szf 6
        jmp oc9
        stf 6
        plinst (dac  $\overline{s}sa$ 
        lac (dio  $\overline{s}si$ 
        jmp ocd
oc9,    clf 6
        plinst (lac  $\overline{s}sa$ 
        lac (lio  $\overline{s}si$ 
        jmp ocd

```


/display a star

define

```

    starp
    add bx
    swap
    add by
    swap
    ioh
    dpy-4000
    terminate

```

blp, dap blx /star

```

    szs 60
    jmp blx
    random
    rar 9s
    and (add 340
    spa
    xor (377777
    dac bx
    lac ran
    ral 4s
    and (add 340
    spa
    xor (377777
    dac by
    jsp bpt
    ioh

```

blx, jmp .

bpt, dap bpx
 random
 sar 9s
 sar 5s
 spa
 cma
 sal 3s
 add (bds
 dap bjm
 cla cli clf 6-opr-opr
 dpy-4000

bjm, jmp .

bds, repeat 20, starp
 szf 6

bpx, jmp .
 stf 6
 cma
 swap
 cma
 swap
 jmp bjm

/background display • 3/13/62, prs.

```

define
dislis J, Q, B
repeat 6, B=B+B
clf 5
lac flo+R
dap fpo+R
fs,    dap fin+R
        dap fyn+R
        idx fyn+R

fin,    lac      /lac X
        sub fpr  /right margin
        sma
        jmp fgr+R
        add (2000

frr,    spq
fou,    jmp fuu+R
fie,    sub (1000
        sal 8s
fyn,    lio      /lio Y
        dpy-i+B
        stf 5
fid,    idx fyn+R
        sad (lio Q+2
        jmp flp+R
        sad fpo+R
        jmp fx+R
        dap fin+R
        idx fyn+R
        jmp fin+R

fgr,    add (-20000+2000
        jmp frr+R

fuu,    szf 5
fx,     jmp flo+R+1      /return
        idx flo+R
        idx flo+R
        sas (Q+2
        jmp fid+R
        law J
        dac flo+R
        jmp fid+R

flp,    lac (lio J
        sad fpo+R
        jmp fx+R
        dap fin+R
        law J+1
        dap fyn+R
        jmp fin+R

fpo,    lio
flo,    J
        terminate

```

```
define
background
    jsp bck
    termin

bck,    dap bck
        szs 40
        jmp bck
        isp bcc

bck,    jmp .
        law i 2
        dac bcc
        dislis 1j,1q,3
        dislis 2j,2q,2
        dislis 3j,3q,1
        isp bkc
        jmp bck
        law i 20
        dac bkc
        law i 1
        add fpr
        spa
        add (20000
        dac fpr
        jmp bck

bcc,    0
bkc,    0
fpr,    10000
```

mul=mus
div=dis

start

spacewar 4.0ts 5/4/63 ddp •pt 2

/main control routine for spaceships

```

nob=30                                /total number of colliding objects

m10,      load mtc, -4000              /delay for loop
          init m11, mtb               /loc of calc routines
          add (nob
          dap mx1                      / x
nx1=mtb nob
          add (nob
          dap my1                      / y
ny1=nx1 nob
          add (nob
          dap ma1                      / count for length of explosion or torp
na1=ny1 nob
          add (nob
          dap mb1                      / count of instructions taken by calc routine
nb1=na1 nob
          add (nob
          dac mdx                      / dx
ndx=nb1 nob
          add (nob
          dac mdy                      / dy
ndy=ndx nob
          add (nob
          dap mom                      /angular velocity
nom=ndy nob
          add (2
          dap mth                      / angle
nth=nom 2
          add (2
          dac mfu                      /fuel
nfu=nth 2
          add (2
          dac mtr                      / no torps remaining
ntr=nfu 2
          add (2
          dap mot                      / outline of spaceship
not=ntr 2
          add (2
          dap mco                      / old control word
nco=not 2
          add (2
          dac mh1
nh1=nco 2
          add (2
          dac mh2
nh2=nh1 2
          add (2
          dac mh3
nh3=nh2 2
          add (2
          dac mh4
nh4=nh3 2
nnn=nh4 2
          law ss1
          xor mtb
          sza
          jmp mdn
          law ss2
          xor mtb 1

```

```
      sza
      jmp mdn
      law 1      / test if both ships out of torps
      add ntr
      spa
      jmp md1
      law 1
      add ntr 1
      spa i
      jmp mdn
md1,   xct tlf    / restart delay is 2X torpedo life
      sal 1s
      dac ntd
      jmp ml1

mdn,   count ntd,ml1
      stf 1
      stf 2
      law ss1
      xor mtb
      sza
      clf 1
      sza i
      idx 1sc
      law ss2
      xor mtb 1
      sza
      clf 2
      sza i
      idx 2sc
      clf 2
      jmp a
```

```

a1,      law mg2          / test word control
         dac cwg
         jmp a

a40,     law cwr      / here from start at 4
         dac cwg
         jmp a6

a,       lac gct
         sma
         jmp a5
         count gct, a5
         lac isc
         sas sc
         jmp a4
         law i 1
         dac gct

a5,      lat
         and (40
         sza i
         jmp a2

a4,      lac isc
         llo sc
         hlt
         lat
         and (40
         sza
         jmp a2
         dzm isc
         dzm sc

a6,      lat
         rar 6s
         and (37
         sza
         cma
         dac gct

a2,      clear mtb, nnn-1  / clear out all tables
         law ss1
         dac mtb
         law ss2
         dac mtb 1
         lac (200000
         dac nx1
         dac ny1
         cma
         dac nx1 1
         dac ny1 1
         lac (144420
         dac nth

```

```

a3,      law nnn                / start of outline program
         dac not
         lio ddd
         spi i
         jmp a3
         jda oc                / compile outline
         ot1
         dac not 1
         jda oc
         ot2
         xct tno
         dac ntr
         dac ntr 1
         lac foo
         dac nfu
         dac nfu+1
         law 2000
         dac nb1
         dac nb1 1
         xct mhs
         dac nh2
         dac nh2 1
         jmp ml0

```


/ control word get routines

```

mg1,      dap mg3
          cli
          iot 11
          rir 4s
mg3,      jmp .

mg2,      dap mg4
          lat
          swap
mg4,      jmp .

ml1,      lac .                / 1st control word
          sza i                / zero if not active
          jmp mq1              / not active
          swap
          idx moc
          spi
          jmp mq4
          law 1
          add ml1
          dap ml2
          law 1
          add mx1
          dap mx2
          law 1
          add my1
          dap my2
          law 1
          add ma1
          dap ma2
          law 1
          add mb1
          dap mb2

mot,      lac .
          dap sp5
ml2,      lac .                / 2nd control word
          spq                  / can it collide?
          jmp mq2              / no
mx1,      lac .                / calc if collision
mx2,      sub .                / delta x
          spa                  / take abs val
          cma
          dac mt1
          sub me1              / < EPSILON ?
          sma
          jmp mq2              / no
my1,      lac .
my2,      sub .
          spa
          cma
          sub me1              / < epsilon ?
          sma
          jmp mq2              / no
          add mt1
          sub me2
          sma
          jmp mq2
          lac (mex 400000      / yes, EXPLODE
          dac i ml1           / replace calc routine with explosion
          dac i ml2
          lac i mb1           / duration of explosion

```

```

mb2,      add .
          cma
          sar 8s
          add (1
ma1,      dac .
ma2,      dac .
mq2,      idx mx2          / end of comparison loop
          idx my2
          idx ma2
          idx mb2
          index ml2, (lac mtb nob, ml2

```

```

mq4,      lac i ml1          / routine for calculating spaceship
          dap . 1           / or other object and displaying it
          jsp .
mb1,      lac .             / alter count of number of instructions
          add mtc
          dac mtc
mq1,      idx mx1           / end of comparison and display loop
          idx my1
          idx ma1
          idx mb1
          idx mdx
          idx mdy
          idx mom
          idx mth
          idx mfu
          idx mtr
          idx mot
          idx mco
          idx mh1
          idx mh2
          idx mh3
          idx mh4
          index ml1, (lac mtb nob-1, ml1
          lac i ml1         / display and compute last point
          sza i             / if active
          jmp mq3
          dap . 1
          jsp .
          lac i mb1
          add mtc
          dac mtc
mq3,      background       / display stars of the heavens
          jsp blp          / display massive star
          count mtc, .     / use up rest of time of main loop
          jmp ml0         / repeat whole works

```

/ misc calculation routines

/ explosion

```

mex,      dap mxr
          lac i mdx
          sar 3s
          add i mx1
          dac i mx1
          lac i mdy
          sar 3s
          add i my1
          dac i my1
          law mst
          dap msh
          lac i mb1
          cma cli-opr
          sar 3s
          dac mxc
ms1,      sub (140
          sma
          idx msh
mz1,      random
          and (777
          ior (scl
          dac m11
          random
          scr 9s
          sir 9s
msh,      xct .
m11,      hlt
          add i my1
          swap
          add i mx1
          dpy-i 300
          count mxc, mz1
          count i ma1, mxr
          dzm i m11
mxr,      jmp .

mst,      scr 1s
          scr 3s

```

/ time involved

/ torpedo calc routine

```

tcr,      dap trc
          count i ma1, tc1
          lac (mex 400000
          dac i m11
          law i 2
          dac i ma1
          jmp trc

tc1,      lac i mx1
          sar 9s
          xct the
          add i mdy
          dac i mdy

```

```
    sar 3s
    add i my1
    dac i my1
    sar 9s
    xct the
    add i mdx
    dac i mdx
    sar 3s
    add i mx1
    dac i mx1
    dispt i, i my1, 1
    jmp .
trc,
```

/ hyperspace routines

SPW-22

/ this routine handles a non-colliding ship invisibly
/ in hyperspace

```
hp1,      dap hp2
          count i ma1, hp2
          law hp3          / next step
          dac i ml1
          law 7
          dac i mb1
          random
          scr 9s
          sir 9s
          xct hr1
          add i mx1
          dac i mx1
          swap
          add i my1
          dac i my1
          random
          scr 9s
          sir 9s
          xct hr2
          dac i mdy
          dio i mdx
          lac ran
          dac i mth
          random
          sar 6s
          dac i mom
hp4,      lac i mth
          sma
          sub (311040
          spa
          add (311040
          dac i mth
          xct hd2
          dac i ma1
hp2,      jmp .
```

/ this routine handles a ship breaking out of
/ hyperspace.

```
hp3,      dap hp5
          count 1 ma1, hp6
          lac i mh1
          dac i ml1
          law 2000
          dac i mb1
          count 1 mh2, hp7
          dzm i mh2
```

```
hp7,      xct hd3
           dac i mh3
           lac i mh4
           add hur
           dac i mh4
           random
           ior (400000
           add i mh4
           spa
           jmp hp5
           lac (mex 400000
           dac i ml1
           law i 10
           dac i ma1
           law 2000
           dac i mb1
hp6,      lac i mx1
           dispt i, i my1, 2
hp5,      jmp .
```

/ spaceship calc

```

ss1,      dap srt           / first spaceship
          jsp i  $\overline{\text{cwg}}$ 
          dio  $\overline{\text{scw}}$ 
          jmp sr0

ss2,      dap srt           / second spaceship
          jsp i  $\overline{\text{cwg}}$ 
          rir 4s
          dio  $\overline{\text{scw}}$ 

sr0,

sc1,      lio  $\overline{\text{scw}}$          /control word
          clf 6 cla-opr     /update angle
          spi
          add maa
          ril 1s
          spi
          sub maa

mom,      add .
          dac i mom
          szs 10
          jmp sr8
          dzm i mom
          ral 7s

sr8,      ril 1s
          spi
          stf 6
          lio i  $\overline{\text{mfu}}$ 
          spi i
          clf 6

mth,      add .
          sma
          sub (311040
          spa
          add (311040
          dac i mth
          jda  $\overline{\text{sin}}$ 
          dac  $\overline{\text{sn}}$ 
          dzm  $\overline{\text{bx}}$ 
          dzm  $\overline{\text{by}}$ 
          szs 60
          jmp bsg
          lac i mx1
          dac  $\overline{\text{t1}}$ 
          mul  $\overline{\text{t1}}$ 
          scr 1s
          dac  $\overline{\text{acx}}$ 
          cla
          scr 2s
          dio  $\overline{\text{lox}}$ 
          lac i my1
          dac  $\overline{\text{t1}}$ 
          mul  $\overline{\text{t1}}$ 
          scr 1s
          dac  $\overline{\text{acy}}$ 

```



```

cla
scr 2s
swap
add  $\bar{t}ox$ 
swap
scl 2s
add  $\bar{a}cx$ 
add  $\bar{a}cy$ 
sub str
sma i sza-skp
jmp pof
add str
varsft
dac  $\bar{t}1$ 
jda  $\bar{s}qt$ 
mul  $\bar{t}1$ 
undosft
scr 9s
scr 6s
szs i 20          / switch 2 for light star
scr 2s
sza
jmp bsg
scr 1s
dio  $\bar{t}1$ 
integrate mx1,  $\bar{b}x$ 
integrate my1,  $\bar{b}y$ 
bsg,
cla
sad i mfu
clf 6
lac i mth
jda  $\bar{c}os$ 
dac  $\bar{c}s$ 
sar 9s
xct sac
szf i 6
cla
add  $\bar{b}y$ 
diff  $\bar{m}dy$ , my1, (sar 3s
lac  $\bar{s}n$ 
sar 9s
xct sac
cma
szf i 6
cla
add  $\bar{b}x$ 
diff  $\bar{m}dx$ , mx1, (sar 3s
sp1,
scale  $\bar{s}n$ , 5s,  $\bar{s}sn$ 
sp2,
scale  $\bar{c}s$ , 5s,  $\bar{s}cn$ 
lac i mx1

```

```

sub ssn
dac sx1
sub ssn
dac stx
lac i my1
add scn
dac sy1
add scn
dac sty
scale sn, 9s, ssn
scale cs, 9s, scn
lac ssn
dac ssm
add scn
dac ssc
dac ssd
lac ssn
sub scn
dac csn
cma
dac csm
lac scn
dac scm
cla cli-opr
dpy-4000
sp5, jmp .
sq6, loh
      ranc sar 9s, sar 4s, src
      lio scw
      ril 2s
      spi i / not blasting
      jmp sq9 / no tail
sq7, scale sn, 8s, ssn
      scale cs, 8s, scn
      count 1 mfu, st2
      dzm 1 mfu
      jmp sq9

st2, yincr sx1, sy1, sub
      dispt 1, sy1
      count src, sq7
sq9, count 1 ma1, sr5 / check if torp tube reloaded
      dzm 1 ma1 / prevent count around
mco, lac . / previous control word
      cma
      szs 1 30
      clc
      and scw / present control word
      ral 3s / torpedo bit to bit 0
      sma
      jmp sr5 / no launch
      count 1 mtr, st1 / check if torpedos exhausted
      dzm 1 mtr / prevent count around
      jmp sr5

st1, init sr1, mtb / search for unused object
sr1, lac .
      sza 1 / 0 if unused
      jmp sr2
      index sr1, (lac mtb nob, sr1
      hlt / no space for new objects
      jmp .-1

```

```

sr2,      lac (tcr           / set up torpedo calc
          dac i sr1
          law nob
          add sr1
          dap ss3
          lio stx
ss3,      dio .
          add (nob
          dap ss4
          lio sty
ss4,      dio .
          add (nob
          dap sr6
          add (nob
          dap sr7
          add (nob
          dap sr3
          add (nob
          dap sr4
          lac sn
          xct tvl
          cma
          add i mdx
sr3,      dac .
          lac cs
          xct tvl
          add i mdy
sr4,      dac .
          xct rlt
          dac i ma1           / permit torp tubes to cool
trf,      xct tlf / life of torpedo
sr6,      dac .
          law 20
sr7,      dap .               / length of torp calc.
sr5,      count i mh3, st3    / hyperbutton active?
          dzm i mh3
          lac i mh2
          sza i
          jmp st3
          lac scw
          cma
          ior i mco
          and (600000
          sza
          jmp st3
          lac i ml1
          dac i mh1
          lac (hp1 400000
          dac i ml1
          xct hd1
          dac i ma1
          law 3
          dac i mb1
st3,
srt,      jmp .

```

/ here to handle spaceships dragged into star

/ spaceship in star

```
pof,      dzm i mdx
           dzm i mdy
           szs 50
           jmp po1
           lac (377777
           dac i mx1
           dac i my1
           lac i mb1
           dac ssn
           count ssn, .
           jmp srt
```

```
po1,      lac (mex 400000      / now go bang
           dac i m11
           law i 10
           dac i ma1
           jmp srt
```

• 5/

• 5/

/ space for patches

```
mtb,           / table of objects and their properties
```

start 4

stars by prs for s/w 2b

SPUR 31

6077/

/stars 1 • 3/13/62, prs.

decimal

define

mark X, Y

repeat 8, Y=Y+Y

8192-X Y

terminate

| | | |
|-----|-----------------|----------------------|
| 1j, | mark 1537, 371 | /87 Taur, Aldebaran |
| | mark 1762, -189 | /19 Orio, Rigel |
| | mark 1990, 168 | /58 Orio, Betelgeuze |
| | mark 2280, -377 | /9 CMaj, Sirius |
| | mark 2583, 125 | /10 CMin, Procyon |
| | mark 3431, 283 | /32 Leon, Regulus |
| | mark 4551, -242 | /67 Virg, Spica |
| | mark 4842, 448 | /16 Boot, Arcturus |
| 1q, | mark 6747, 196 | /53 Aqil, Altair |
| 2j, | mark 1819, 143 | /24 Orio, Bellatrix |
| | mark 1884, -29 | /46 Orio |
| | mark 1910, -46 | /50 Orio |
| | mark 1951, -221 | /53 Orio |
| | mark 2152, -407 | / 2 CMaj |
| | mark 2230, 375 | /24 Gemi |
| | mark 3201, -187 | /30 Hyda, Alphard |
| | mark 4005, 344 | /94 Leon, Denebola |
| 2q, | mark 5975, 288 | /55 Ophi |
| 3j, | mark 46, 333 | /88 Pegs, Algenib |
| | mark 362, -244 | /31 Ceti |
| | mark 490, 338 | /99 Pisc |
| | mark 566, -375 | /52 Ceti |
| | mark 621, 462 | / 6 Arie |
| | mark 764, -78 | /68 Ceti, Mira |
| | mark 900, 64 | /86 Ceti |
| | mark 1007, 84 | /92 Ceti |
| | mark 1243, -230 | /23 Erid |
| | mark 1328, -314 | /34 Erid |
| | mark 1495, 432 | /74 Taur |
| | mark 1496, 356 | /78 Taur |
| | mark 1618, 154 | / 1 Orio |
| | mark 1644, 52 | / 8 Orio |
| | mark 1723, -119 | /67 Erid |
| | mark 1755, -371 | / 5 Leps |
| | mark 1779, -158 | /20 Orio |
| | mark 1817, -57 | /28 Orio |
| | mark 1843, -474 | / 9 Leps |
| | mark 1860, -8 | /34 Orio |
| | mark 1868, -407 | /11 Leps |
| | mark 1875, 225 | /39 Orio |
| | mark 1880, -136 | /44 Orio |
| | mark 1887, 480 | /123 Taur |
| | mark 1948, -338 | /14 Leps |
| | mark 2274, 296 | /31 Gemi |
| | mark 2460, 380 | /54 Gemi |

| | | |
|-----------------|-----|---------------|
| mark 2470, 504 | /55 | Gem1 |
| mark 2513, 193 | /3 | CMin |
| mark 2967, 154 | /11 | Hyda |
| mark 3016, 144 | /16 | Hyda |
| mark 3424, 393 | /30 | Leon |
| mark 3496, 463 | /41 | Leon, Algieba |
| mark 3668, -357 | /nu | Hyda |
| mark 3805, 479 | /68 | Leon |
| mark 3806, 364 | /10 | Leon |
| mark 4124, -502 | /2 | Corv |
| mark 4157, -387 | /4 | Corv |
| mark 4236, -363 | /7 | Corv |
| mark 4304, -21 | /29 | Virg |
| mark 4384, 90 | /43 | Virg |
| mark 4421, 262 | /47 | Virg |
| mark 4606, -2 | /79 | Virg |
| mark 4721, 430 | /8 | Boot |
| mark 5037, -356 | /9 | Libr |
| mark 5186, -205 | /27 | Libr |
| mark 5344, 153 | /24 | Serp |
| mark 5357, 358 | /28 | Serp |
| mark 5373, -71 | /32 | Serp |
| mark 5430, -508 | /7 | Scor |
| mark 5459, -445 | /8 | Scor |
| mark 5513, -78 | /1 | Ophi |
| mark 5536, -101 | /2 | Ophi |
| mark 5609, 494 | /27 | Herc |
| mark 5641, -236 | /13 | Ophi |
| mark 5828, -355 | /35 | Ophi |
| mark 5860, 330 | /64 | Herc |
| mark 5984, -349 | /55 | Serp |
| mark 6047, 63 | /62 | Ophi |
| mark 6107, -222 | /64 | Ophi |
| mark 6159, 217 | /72 | Ophi |
| mark 6236, -66 | /58 | Serp |
| mark 6439, -483 | /37 | Sgtr |
| mark 6490, 312 | /17 | Aqil |
| mark 6491, -115 | /16 | Aqil |
| mark 6507, -482 | /41 | Sgtr |
| mark 6602, 66 | /30 | Aqil |
| mark 6721, 236 | /50 | Aqil |
| mark 6794, 437 | /12 | Sgte |
| mark 6862, -25 | /65 | Aqil |
| mark 6914, -344 | /9 | Capr |
| mark 7014, 324 | /6 | Dlph |
| mark 7318, -137 | /22 | Aqar |
| mark 7391, 214 | /8 | Pegs |
| mark 7404, -377 | /49 | Capr |
| mark 7513, -18 | /34 | Aqar |
| mark 7539, 130 | /26 | Pegs |
| mark 7644, -12 | /55 | Aqar |
| mark 7717, 235 | /42 | Pegs |
| mark 7790, -372 | /76 | Aqar |
| mark 7849, 334 | /54 | Pegs, Markab |

start 4

SPW - 33